

REMARKS

Claims 1-2 and 4-9 remain in the present application. Claims 1-2, 4-5 and 8-9 were amended in this response. No new matter was introduced as a result of the amendments. Support for the present amendments may be found in the amended specification page 1, lines 5-18, page 3, lines 6-12, 20-29 and page 4, lines 1-23.

Claims 1-2 and 4-9 were rejected under 35 U.S.C. §103(a) as being unpatentable over *Novak et al.* (US Patent 6,393,419) in view of *Landresse* (US Patent 6,351,744). Applicant traverse this rejection. Favorable reconsideration is respectfully requested.

Applicant submits that the prior art, alone or in combination, fails to teach or suggest object-oriented applications as well as “a second interface to an access unit which provides different access mechanisms for respectively different memory structures, wherein the unit provides the object-oriented application with appropriate data by accessing the access unit and affords changes to the data independent of changes made to the data via the first interface; and a third interface to a consistency module for automatically updating changes to the data in further object-oriented applications accessing the same data” as recited in claim 1 and similarly in claim 9.

As an initial matter, Applicant notes that the present claims now recite the feature of “object-oriented applications” and is supported in the amended specification on page 1, lines 1-22. It is acknowledged in the specification, and known to those having ordinary skill in the art, that object-oriented programming is based on computer programs that are composed of a collection of individual units, or *objects*, that act on each other, as opposed to a traditional view (such as in *Novak*, see col. 3, lines 38-51) in which a program is a collection of functions or procedures, or simply as a list of instructions to the computer. Generally, each object in object-oriented programming is capable of receiving messages, processing data, and sending messages to other objects. Such a configuration is neither taught nor suggested in *Novak*. Regarding the Office Action’s assertion that *Novak* teaches “objects” (see col. 3, lines 29-52), Applicant respectfully submits that this language refers to the “objects of the invention,” rather than a specific object to be utilized by an object-oriented application.

Furthermore, the amended claims recite “different access mechanisms for respectively different memory structures.” Under the teaching of *Novak*, the memory structures are disclosed

as having a single access mechanism for all memory structures (col. 2, lines 35-47; col. 4, lines 12-29).

Also, Novak fails to teach or suggest a third interface to a consistency module for automatically updating changes to the data in further object-oriented applications accessing the same data. The amended claim clarifies the point that an object-oriented application accessing the same centrally-stored data will re-load the data containing the updates in each case. As argued previously, *Novak* teaches a database using a change counter value which tracks the modification status of various database records to avoid a data corruption situation and thus allows multiple clients to access and save records to a database without data corruption (col. 3, lines 29-37). *Novak* discloses the use of a change counter to synchronize databases to avoid corruption when client A saves a newly modified version of record X, thereby over-writing client B's previously saved modifications. The database corruption situation is disclosed as occurring whenever multiple database clients simultaneously edit and save portions of the same database data, unbeknownst to each other (col. 4, lines 38-44). The data corruption is averted in *Novak* by comparing a change counter value (CCV) to a previous value, along with the access authorization granted to the user desiring changes to the document (col. 5, lines 1-6; 35-50; 60-67). Thus, *Novak* merely discloses a mechanism through which a second application is prevented from writing to a set of data which was changed by a first application. In cases where a further application attempts to make modifications to the data, the further applicant must manually contact a database manager to obtain permission to effect changes in the data (col. 8, lines 40-56). Also, Novak only discloses that the changes are only affected to the centrally-stored document – the “applications” accessing the document are not affected.

Regarding *Landresse*, the reference also does not teach object-oriented applications, and instead relies on data sets and logical structures to execute conventional program instructions (“routines” - col. 9, lines 32-44, col. 9, line 57 – col. 10, line 18; col. 10, line 66 – col. 11, line 60). Furthermore *Landresse* does not teach “different access mechanisms for respectively different memory structures, wherein the unit provides the application with appropriate data by accessing the access unit and affords changes to the data independent of changes made to the data via the first interface” as recited in claim 1 and similarly recited in claim 9. *Landresse* discloses a plurality of processors and database engines that are inter-related through a memory

system to provide an asynchronous mechanism for distributing the operating workload of flushing data buffers over a multiple number (N) of worker tasks (col. 2, lines 35-39). *Landresse* further discloses that a multiple number of database engines having access routines available to multiple numbers of user application programs are connected to different sets of data file structures in databases composed of multiple physical files (col. 2, lines 39-42). The advantage of this configuration according to *Landresse* is to make use of the user tasks already running on multiple processors (CPUs) so that the time required to accomplish any particular one of the given tasks is minimized in several specialized ways (col. 2, lines 42-47). While *Landresse* discloses different memory structures, nothing in the disclosure teaches or suggests providing an object-oriented application with appropriate data by accessing the access unit and affords changes to the data independent of changes made to the data via the first interface.

Furthermore, Applicant submits that there is no teaching, suggestion or motivation for one of ordinary skill in the art to combine the *Novak* and *Landresse* references in the manner suggested in the Office Action. In making a determination that an invention is obvious, the Patent Office has the initial burden of establishing a *prima facie* case of obviousness. *In re Rijckaert*, 9 F.3d 1531, 1532, 28 U.S. P.Q.2d 1955, 1956 (Fed. Cir. 1993). "If the examination at the initial stage does not produce a *prima facie* case of unpatentability, then without more the applicant is entitled to grant of the patent." *In re Oetiker*, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992).

The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991). The initial burden is on the examiner to provide some suggestion of the desirability of doing what the inventor has done. "To support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or impliedly suggest the claimed invention or the examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the references." *Ex parte Clapp*, 227 USPQ 972, 973 (Bd. Pat. App. & Inter. 1985). When the motivation to combine the teachings of the references is not immediately apparent, it is the duty of the examiner to explain

why the combination of the teachings is proper. *Ex parte Skinner*, 2 USPQ2d 1788 (Bd. Pat. App. & Inter. 1986). (see MPEP 2142).

Further, the Federal Circuit has held that it is “impermissible to use the claimed invention as an instruction manual or ‘template’ to piece together the teachings of the prior art so that the claimed invention is rendered obvious.” *In re Fritch*, 23 U.S.P.Q.2d 1780, 1784 (Fed. Cir. 1992). “One cannot use hindsight reconstruction to pick and choose among isolated disclosures in the prior art to deprecate the claimed invention” *In re Fine*, 837 F.2d 1071 (Fed. Cir. 1988).

Moreover, the Federal Circuit has held that “obvious to try” is not the proper standard under 35 U.S.C. §103. *Ex parte Goldgaber*, 41 U.S.P.Q.2d 1172, 1177 (Fed. Cir. 1996). “An-obvious-to-try situation exists when a general disclosure may pique the scientist curiosity, such that further investigation might be done as a result of the disclosure, but the disclosure itself does not contain a sufficient teaching of how to obtain the desired result, or that the claim result would be obtained if certain directions were pursued.” *In re Eli Lilly and Co.*, 14 U.S.P.Q.2d 1741, 1743 (Fed. Cir. 1990).

As was argued above, *Novak* discloses a database processing system that prevents secondary users from changing or manipulating inventory or sales data, where user authorization and CCV processing restricts access to the database. *Landresse* is centered on making use of the user tasks already running on multiple processors so that the time required to accomplish any particular one of the given tasks is minimized. Applicant submits that there is no teaching, suggestion or motivation for one having ordinary skill in the art to rely on the multi-processor configuration of *Landresse* in light of the teaching in *Novak*. Furthermore, the buffer pool of *Landresse* is completely inapplicable to *Novak* and teaches away from the reference, as it is the express purpose to the *Landresse* to perform simultaneous application tasks among the I/O buffer structures in an asynchronous manner (col. 1, line 65- col. 2, line 7). In contrast, *Novak* clearly teaches that user changes to documents are to be performed in a sequential manner to avoid data corruption and expressly relies on synchronous operation (col. 4, lines 37-44). For at least these reasons, Applicant respectfully submits the rejection is improper and should be withdrawn.

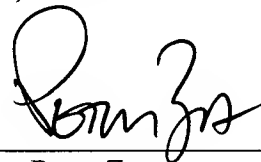
In light of the above, Applicant respectfully submits that independent claim 1 and 9 of the present application, as well as claims 2 and 4-8 which respectfully depend therefrom, are both

novel and non-obvious over the art of record. Accordingly, Applicant respectfully requests that a timely Notice of Allowance be issued in this case.

Applicant further submits that no additional fees are due in connection with this Response at this time. However, if any additional fees are due in connection with this application as a whole, the Examiner is authorized to deduct said fees from Deposit Account No.: 02-1818. If such a deduction is made, please indicate the attorney docket number (0112740-357) on the account statement.

Respectfully submitted,

BELL, BOYD & LLOYD LLC

BY 

Peter Zura

Reg. No. 48,196

Customer No.: 29177

(312) 807-4208

Dated: February 21, 2006